Assignment (5)

24789 - Intermediate Deep Learning (Spring 2024)

Out Date: 2024/4/16 (Tue)
Due Date: 2024/4/24 (Wed) @ 11:59 pm EST

All exercises should be submitted to Gradescope. There are 2 assignments in Gradescope for this homework, one where you will submit a PDF of your answers and another where you will submit a zipped file containing your code and any additional files (including the PDF of your submission) as supplemental material. Please make sure to include all relevant code and plots in the PDF. Please have a new page for each solution, and box your answers (if applicable). Please use Piazza for any questions about the assignment. Also, **please start early** as diffusion models can take a while to train.

Programming Exercises (50 points)

PROBLEM 1

Contrastive Learning Using SimCLR (50 points)

In this assignment, you are asked to train a SimCLR model for contrastive learning on the CIFAR-10 dataset using the provided starter code. The SimCLR model is a self-supervised learning framework that learns visual representations by contrasting positive and negative pairs of augmented images.

1a) Complete the implementation of the ***nt_xent*** loss function, which computes the normalized temperature-scaled cross entropy loss for SimCLR. Given below is the equation for the loss function:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\boldsymbol{z}_i, \boldsymbol{z}_j\right)/\tau\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\text{sim}\left(\boldsymbol{z}_i, \boldsymbol{z}_k\right)/\tau\right)} \tag{1}$$

1b) Complete the implementation of the ***barlow_twins*** loss function, which computes the normalized temperature-scaled cross entropy loss for SimCLR. Given below is the equation for the loss function:

Cross correlation matrix

$$\mathcal{C}_{ij} \triangleq \frac{\sum_b z_{b,i}^A z_{b,j}^B}{\sqrt{\sum_b \left(z_{b,i}^A\right)^2} \sqrt{\sum_b \left(z_{b,j}^B\right)^2}} \tag{2}$$

Barlow Twins loss

$$\mathcal{L}_{\mathcal{BT}} \triangleq \sum_i \left(1 - \mathcal{C}_{ii}\right)^2 + \lambda \quad \sum_i \sum_{j \neq i} \mathcal{C}_{ij}^2 \tag{3}$$

2) Define the forward function for the SimCLR model. It uses resnet18 as the base encoder. The forward function should return both the output from the encoder and the output from the projection head

3) Define the transformations, train dataset and dataloader, model, optimizer, scheduler, and parameters required for training

4) Fill in the missing parts in train function and pass in the appropriate hyperparameters. Train the model and save ***simclr_best_epoch.pt***. You will need it for the fine-tuning

5) Define the train and test transformations, train and test dataset and dataloader, model, optimizer, scheduler, and parameters required for finetuning

6) Run the fine-tuning and save ***simclr_lin_best.pt*** and plot the train and test loss and accuracies wrt. epochs and add the plots in your report.

A template code and datasets are been given for you to start, but you are not required to follow the code.

You should submit your code and your best model after finetuning and the report should be the pdf version of hw5.ipynb. Make sure you include the following details in your report:

- train and test plots for accuracies and loss for fine-tuning
- the hyperparameters for both training and fine-tuning
- data augmentations used for both training and fine-tuning
- the final test and train accuracies after fine-tuning

To get full credits, the final accuracy on **test set** should be more than 75%. You will get 5 bonus credits if the **accuracy** on the test set is more than 85%.